

Modelagem de Banco de Dados

Prof.^a Ma. Jessica Oliveira

Aula 07 – 09/04/2025

Normalização Avançada (BCNF, 4FN e 5FN).

Revisão das Formas Normais Básicas.

(1FN, 2FN e 3FN)

Conceitos fundamentais de Normalização.

- É um processo de organização das tabelas e colunas para **reduzir redundâncias e prevenir anomalias** de inserção, exclusão e atualização.
- As anomalias ocorrem quando os dados estão estruturados de forma que atualizações isoladas gerem inconsistências ou duplicações.
- As formas normais (1FN, 2FN, 3FN e as formas mais avançadas) definem regras progressivas para minimizar esses problemas.

Conceitos fundamentais de Normalização.

- Imagine que você está arrumando seu guarda-roupa. Se você joga tudo de qualquer jeito lá dentro, vai ser difícil encontrar o que precisa, e pode até guardar coisas repetidas sem perceber.
- Normalizar é como organizar esse guarda-roupa: separar camisetas, calças, meias... **tudo no seu lugar, sem repetir nada desnecessário.**
- No banco de dados, normalizar é:
 - Organizar as tabelas direito;
 - Evitar informações repetidas;
 - Facilitar a inserção, atualização e remoção de dados, **sem erro ou bagunça.**

1FN: Primeira Forma Normal.

- Uma tabela está na 1FN se todos os atributos (colunas) contêm valores atômicos (não divisíveis) e não existem grupos repetitivos, ou seja: cada **coluna** tem **um único valor por linha** (nada de lista com vírgula);
- Problemas causados quando não está em 1FN:
 - Dificuldade de pesquisar valores;
 - Redundância ou inconsistência.
- Como corrigir: criar colunas de forma que cada atributo seja único e atenda ao critério de atomicidade, logo, nada de repetir colunas com o mesmo tipo de informação.

1 FN: exemplo do que não fazer.

Nome	Telefone1	Telefone2
Fulaninha da Silva	(61) 9 9999-0000	(61) 3333-5555

Nome	Telefones
Fulaninha da Silva	(61) 9 9999-0000, (61) 3333-5555

- Aqui está errado por, na primeira tabela, termos colunas repetidas e, na segunda, termos vários valores em uma mesma célula.

1 FN: exemplo de correção.

- Separar em outra tabela:

ID_Aluno	Nome
001	Fulaninha da Silva

ID_Telefone	ID_Aluno	Numero
0001	001	(61) 9 9999-0000
0002	001	(61) 3333-5555

2FN: Segunda Forma Normal.

- Uma tabela está na 2FN se, além de atender à 1FN, **todos os atributos não-chave dependem funcionalmente da chave primária inteira** (não pode haver dependência parcial de apenas parte de uma chave composta).
- Ou seja: todos os dados da tabela **dependem da chave primária inteira, e não apenas de uma parte dela** (isso vale se a chave primária tiver mais de uma coluna).

2FN: exemplo do que não fazer.

ID_Pedido	ID_Lanche	Nome_Lanche
0102	3035	Pizza de Calabresa

- “Nome_Lanche” depende só do “ID_Lanche”, não depende do pedido.
- Isso é problema porque:
 - Se o lanche aparecer em vários pedidos, o nome se repetirá;
 - Mais chances de digitar algo errado.

2FN: exemplo de correção.

- Separar em duas tabelas:

Tabela Lanche:

ID_Lanche	Nome_Lanche
3035	Pizza de Calabresa

Tabela Itens_Pedido:

ID_Pedido	ID_Lanche
0102	3035

3FN: Terceira Forma Normal.

- Uma tabela está na 3FN se, além de atender à 2FN, **todos os atributos não-chave dependem somente da chave primária e não de outros atributos não-chave** (eliminando dependências transitivas).
- Ou seja: os dados da tabela **dependem apenas da chave primária**, e não de outros dados que também não são chave.

3FN: exemplo do que não fazer.

ID_Funcionario	Nome	Empresa	Cidade_Empresa
0047	Siclano Vieira	BabyCorp	Brasília

- Aqui, “Cidade_Empresa” depende da “Empresa”, não do funcionário.
- Problema:
 - Se mudar a cidade da empresa, pode ter que alterar várias linhas;
 - Pode causar erros ou dados desatualizados.

3FN: exemplo de correção.

Tabela Empresa:

Nome_Empresa	Cidade
BabyCorp	Brasília

Tabela Funcionario:

ID_Funcionario	Nome	Nome_Empresa
0047	Siclano Vieira	BabyCorp

Resumo importante!

- **1FN** elimina **repetições** e **valores não atômicos**.
- **2FN** elimina **dependências parciais** (atributos que dependem de apenas parte da chave primária).
- **3FN** elimina **dependências transitivas** (atributos não-chave que dependem de outros não-chave).

Introdução às Formas Normais Avançadas.

(BCNF, 4FN e 5FN)

BCNF: Boyce-Codd Normal Form.

- Uma tabela está em BCNF se, para toda dependência funcional $X \rightarrow Y$, X é uma chave candidata.
- Em outras palavras, sempre que um conjunto de atributos (X) determina outro atributo (Y), aquele conjunto (X) deve ser capaz de identificar univocamente a tupla, ou seja, deve ser uma chave (ou chave candidata).
- Ou seja... Toda relação entre colunas (dependência funcional) acontece **somente se quem determina (lado esquerdo da seta) for uma chave candidata.**

BCNF: exemplo do que não fazer.

Docente	Disciplina	Sala
Beltrano Souza	Computação e Sociedade	A111

- Suponha que:
 - Cada matéria tem um único professor;
 - Cada sala só tem um professor por vez.
- Então temos:
 - Matéria → Professor
 - Sala → Professor
- Mas “Matéria” e “Sala” não são chave primária, e estão mandando nos dados. Isso viola a BCNF.

BCNF: exemplo de correção.

Disciplina_Docente

Disciplina	Docente
Computação e Sociedade	Beltrano Souza

Sala_Docente

Sala	Docente
A111	Beltrano Souza

4FN: Quarta Forma Normal.

- Uma tabela está em 4FN se, além de estar em BCNF, **não possui dependências multivaloradas que gerem duplicações.**
- Dependência multivalorada ocorre quando um atributo pode assumir múltiplos valores independentemente de outro atributo.
- Não se trata de uma simples dependência funcional, mas de cenários em que um registro abrange duas ou mais listas de valores independentes.
- Formalmente, temos uma dependência multivalorada $X \twoheadrightarrow Y$ quando, para um determinado valor de X , há um conjunto de valores possíveis de Y que não dependem de outros atributos.

4FN: exemplo do que não fazer.

Projeto	Desenvolvedor	Linguagem
Alpha	Fulana Silva	Java
Alpha	Beltrana Lima	Python
Alpha	Siclano Souza	Java
Alpha	Tetrano Vieira	Python

- O projeto tem vários desenvolvedores e várias linguagens, mas um não depende do outro. Isso causa duplicação.

4FN: exemplo de correção.

Projeto_Desenvolvedor

Projeto	Desenvolvedor
Alpha	Fulana Silva
Alpha	Beltrana Lima
Alpha	Siclano Souza
Alpha	Tetrano Vieira

Projeto_Linguagem

Projeto	Linguagem
Alpha	Java
Alpha	Python

- Se cada desenvolvedor usa qualquer linguagem, e não importa quem usa o quê, a gente separa em duas tabelas: uma de desenvolvedores e outra de linguagens. Se for importante saber quem usou o quê, aí sim a gente cria essa tabela intermediária. O segredo está em **entender se há ou não ligação direta entre os dados.**

5FN: Quinta Forma Normal.

- Uma tabela está em 5FN se, além de atender à 4FN, toda dependência de junção (*join dependency*) é resultado da chave primária.
- Isso significa que qualquer decomposição da tabela em relações menores pode ser feita sem perda de informações e sem introduzir redundâncias.
- Dependência de junção: dizemos que existe uma dependência de junção se uma tabela R pode ser decomposta em R1, R2, R3... de modo que R é a junção exata dessas tabelas menores, sem que haja perda de dados ou duplicação após a junção.

5FN: Quinta Forma Normal.

- Alguns cenários complexos de relacionamentos “muitos para muitos” entre várias entidades podem, em 4FN, ainda apresentar redundâncias sutis que só a verificação de 5FN consegue resolver.
- Resumindo: **se você pode dividir uma tabela em partes menores, e quando junta de novo não perde nada e não cria dados errados, então ela está na 5FN.**

5FN: exemplo do que não fazer.

Discente	Docente	Tema_Trabalho
Fulana Silva	Tetrana Costa	Web Design
Fulana Silva	Fulano Rodrigues	Redes de Computadores
Siclano Souza	Tetrana Costa	Web Design
Siclano Souza	Tetrana Costa	Redes de Computadores

- O que está acontecendo aqui?
 - Fulana Silva pode ter feito dois trabalhos diferentes, um com o tema “Redes de Computadores” e outro “Web Design”.
 - Docentes diferentes avaliaram temas diferentes.
 - Discentes podem ter feito vários temas com vários docentes.

5FN: exemplo do que não fazer.

Discente	Docente	Tema_Trabalho
Fulana Silva	Tetrana Costa	Web Design
Fulana Silva	Fulano Rodrigues	Redes de Computadores
Siclano Souza	Tetrana Costa	Web Design
Siclano Souza	Tetrana Costa	Redes de Computadores

- Temos três coisas separadas:
 - Discentes fazem trabalhos;
 - Docentes avaliam temas;
 - Discentes são avaliados por docentes.

5FN: exemplo do que não fazer.

Discente	Docente	Tema_Trabalho
Fulana Silva	Tetrana Costa	Web Design
Fulana Silva	Fulano Rodrigues	Redes de Computadores
Siclano Souza	Tetrana Costa	Web Design
Siclano Souza	Tetrana Costa	Redes de Computadores

- **Problema:** se essas três informações forem **independentes**, colocar tudo em uma única tabela gera **duplicação de dados**. Por exemplo, se o mesmo tema for avaliado por mais de um professor e mais de um aluno trabalhar com esse tema, as combinações se multiplicam.

5FN: exemplo de correção.

Discente_Docente

Discente	Docente
Fulana Silva	Tetrana Costa
Fulana Silva	Fulano Rodrigues
Siclano Souza	Tetrana Costa

Discente_Tema

Discente	Tema
Fulana Silva	Web Design
Fulana Silva	Redes de Computadores
Siclano Souza	Web Design
Siclano Souza	Redes de Computadores

Docente_Tema

Discente	Tema
Tetrana Costa	Web Design
Tetrana Costa	Redes de Computadores
Fulano Rodrigues	Redes de Computadores

Lembrete importante!



- Nenhuma forma normal é **“mais importante”** que a outra, porque **cada uma resolve um tipo diferente de problema.**
- Elas funcionam como degraus de uma escada: primeiro a gente organiza os dados para não repetir (1FN), depois evita duplicação desnecessária (2FN), depois tira dependências ocultas (3FN), e assim por diante.
- O objetivo da normalização é **deixar o banco de dados mais claro, consistente e fácil de manter.**
- Agora, **não é obrigatório usar todas as formas normais** até a 5FN em todo projeto. Em muitos sistemas reais, chegar até a 3FN ou BCNF já é suficiente.
- A gente só vai além disso (como 4FN ou 5FN) se o sistema for muito grande ou se houver problemas específicos, como listas independentes ou cruzamentos complexos.
- Então, a gente estuda todas para saber o que fazer quando for necessário, mas nem sempre vai usar tudo. **Isso é parte do bom senso do analista.**

Vamos para a prática?



Contexto.

- Uma empresa de TI registra os projetos que desenvolve, com os seguintes dados:
 - Nome do projeto;
 - Tecnologias utilizadas (pode ter várias);
 - Equipe de desenvolvedores (pode ter vários);
 - Cliente para quem o projeto foi feito;
 - Cidade do cliente.

Tabela desnormalizada.

Projeto	Tecnologias	Desenvolvedores	Cliente	Cidade
Sistema Web	HTML, CSS, JavaScript	Ana, Bruno	Criavetando	Brasília
App Financeiro	Kotlin, Firebase	Carla, Bruno, Diego	Organizaê	Uberlândia

- Aplicar todos os passos de normalização, da 1FN à 5FN, e entregar.

Dúvidas?

jessica.oliveira@p.ucb.br